# IMPLEMENTING TRANSFER LEARNING THROUGH PROJECTR AND ITS APPLICATION

by

Gaurav Sharma

A dissertation submitted to Johns Hopkins University in conformity with the requirements for

the degree of Master's in Science and Engineering

Baltimore, Maryland

May 2020

# Abstract

High dimensional datasets are routinely used to answer biological questions and for biological discovery. Dimensionality reduction method applied to such datasets reveal low dimensional latent space that capture both technical and biological effects. Interpretation and validation of these effects is challenging. Transfer learning, a sub-domain of machine learning, can be used for this task. Transfer learning requires a pair of datasets called source dataset where the latent space is learned and target dataset where the latent dimensions are transferred. Biological effects are shared between biologically-related datasets and furthermore, independently-generated datasets do not share technical effects. Thus, transfer learning enables biological validation by evaluating association of learned latent dimensions with annotations of the two datasets. This thesis describes the implementation of a transfer learning method written in R and published as a Bioconductor package called projectR.

**Advisor and Primary Reader:** Elana Fertig

**Thesis Committee:** Seth Blackshaw, Loyal Goff, Luciane T. Kagohara and Elana Fertig

# Acknowledgements

I am incredibly grateful to my advisor Prof. Elana Fertig for her consistent support, guidance and mentorship. I want to thank Prof Seth Blackshaw, Dr Kagohara and Prof Loyal Goff for providing their feedback on this dissertation. I want to also thank Genevieve Stein-O'Brien, Prof Loyal Goff and Prof Elana Fertig for establishing the transfer learning framework and providing guidance in implementation. I want to acknowledge the technical help I received from Thomas Sherman as well.

# Contents

# List of Tables

# List of Figures

# Chapter 1

# Theory

High throughput data generation enabled us to answer system level questions, investigate interactions and identify different biological processes. Depending on the biological question of interest different types of analysis can be performed with the high-dimensional data. However, analysis in higher number of dimensions may not reveal the underlying biological and technical effects captured in the data. Dimensionality reduction methods are used to compute the latent space that capture such effects (Meng et al., 2016). Assigning meaning to these effects learned from some of the dimensionality reduction methods require existing annotations that can be matched with learned dimensions. An annotated latent space is useful for interpreting high dimensional data, even if biological meaning is only known for a subset of dimensions. For example, if the latent dimension captures a transcriptional signature associated with cellular differentiation, then the corresponding assignment of a cell to that dimension can identify how far a cell is located along the differentiation process. While latent space methods can infer a wide range of biological processes in large cellular atlases, smaller experimental datasets may not delineate all sources of biological variation. Thus, it becomes important to design different methods to infer the low dimensional sources of biological variation in new test datasets. One such method is transfer learning. When combined with latent space inference methods for atlas datasets, transfer learning can further relate the biological and technical features learned from the original dataset called the source dataset to a new dataset called the target dataset.

This chapter describes different dimensionality reduction methods to learn latent spaces from single cell data, assignment of biological meaning to latent space dimensions, mathematics of transfer learning and different scenarios where this can be used.

## 1.1 Applications of transfer learning

Transfer learning can be used to gather insights for a variety of applications. For a practical application, consider a pharmaceutical company that wants to conduct a human clinical trial for a drug after gathering drug performance data on mouse. The data gathered from mouse trial can be used to screen human candidates for the trial such that humans who are expected to respond well to the drug are selected. Similarly, humans who are likely to experience adverse events are excluded, ensuring their well-being. This can be done by transferring features learned from source mouse data onto new test human data, generating a cross-species machine learning analysis method. At cellular level, transfer learning can be used to transfer annotations such as cell type, cell state, tissue type, etc. from source molecular dataset to a target molecular dataset.

Apart from transferring annotations in the source data, transfer learning also enables exploration and evaluation of latent spaces learned within the source dataset itself. Notably, transfer learning applied using appropriately annotated target datasets may reveal the importance of some latent dimensions that are not directly measured in the source dataset. For example, previous work by Stein-O'Brien demonstrated that this approach enabled the inference of sex-related differences in gene expression during retinal development in mice that were too young for external genitals to be identified (Stein-O'Brien et al. 2019). Another example of a such application of transfer learning is its application to a transcriptomics source dataset and target

dataset of spatial coordinates may reveal spatial gene expression patterns that cannot be visualized with either of the datasets alone. Such transfer learning can be applied across data modalities, species, tissues, batches, timepoints, and any other scenario where datasets share biological context and were generated independently. Additionally, since the source and target datasets share only biology and not technical artifacts, transfer learning can overcome batch effects and work despite them.

## 1.2 Transfer learning via projection

A fast method to perform transfer learning is to project target dataset on latent space. The idea is to project the target dataset on the latent space learned from source dataset such that the projection minimizes the distance between the target dataset and latent space. The projection method may vary depending on the latent space. This is implemented as an R package called projectR. Figure 1 shows the conceptual framework used for projections implemented in projectR. It shows latent space generated from high dimensional source dataset and a target dataset projected on the latent space.

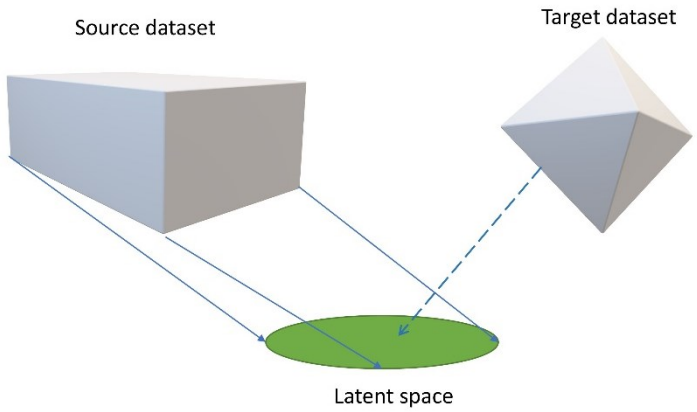


Figure 1.1: Conceptual framework of projectR

These projections provide the weight assigned to target dataset for the biological and technical effects captured by the latent space.

## 1.3 Compatibility of source and target datasets

Transfer learning using using projectR requires that the datasets share some biology. Specifically, as shown in figure 1.2, if the source dataset is X by Y then the target should be X' by Z where X and X' are the same or related dimensions. Table 1.1. provides some examples for them.

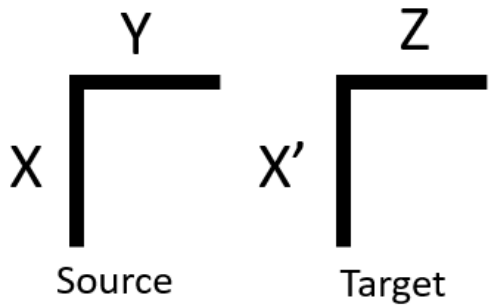


Figure 1.2: Representative figure of source and target datasets

| Source | | Target | |
|---|---|---|---|
| **X** | **Y** | **X'** | **Z** |
| Genes | Species 1 | Genes | Species 2 |
| Genes | Tissue 1 | Genes | Tissue 2 |
| Genes | Tissue 1, Species 1 | Genes | Tissue 2, Species 2 |
| Genes | Samples | Genes | Spatial coordinates |
| Genes | Timepoint 1 | Genes | Timepoint 2 |
| Genes | Samples | Genes | Samples |
| Genes in scRNA-seq | Samples | scATAC-seq | Samples |

Table 1.1: Some examples of compatible source and target datasets

As shown in table 1.1 column X shows the rows of the source dataset. Most of them are genes in a transcriptomics dataset. The last row are genes in a single-cell RNA-sequencing dataset. Column Y and Z denotes the different sample sources or conditions for source dataset and target dataset respectively. X' is similar to X. For scATAC-seq, in the last row of X', the reads around the transcription start site of a gene can be quantified.

## 1.4 Projection on different latent spaces

The method to generate projection varies depending on the method used to generate latent space. The following section discusses different methods to compute latent space and how to compute projection for them.

### 1.4.1 PCA

Principal component analysis (PCA) is a widely used method to compute principal components (PCs) which are orthogonal axes along the direction of maximum variation. If the source data was standardized around zero mean expression of genes, the PCs are computed from the covariance matrix of genes as normalized eigenvectors. Projection places the target data on the PCs defined by the source data. To compute the projections, following approach is used in projectR:

Say the target data matrix is $T$ and $L$ be the gene loadings composed of eigen vectors computed by performing PCA on the source data. First step is to center each row of $T$ around its mean, i.e.,

$$t_{ij} = t_{ij} - \bar{t}_i \tag{1.1}$$

where $t_{ij}$ is the entry in row $i$ and column $j$ of $T$. Equation 1.1 applied to all rows provides us $\overline{T}$. The projected matrix $P$ is computed as shown in equation 1.2

(1.2)

$$P = \bar{T}'L$$

### 1.4.2 NMF

Non-negative matrix factorization (NMF) method decomposes the source data matrix $S$ with genes as rows and samples as columns into amplitude matrix $A$ and pattern matrix $P$.

$$S \sim AP \tag{1.3}$$

Amplitude matrix is genes by patterns and pattern matrix is patterns by samples. Patterns correspond to biological or technical effects captured in the data. For instance, a pattern could correspond to cell type, cell cycle, gender of the sample, donor of the cells, batch, channel, number of genes, etc. The number of patterns is a parameter that is specified before running an NMF algorithm. Coordinated Gene Activity in Pattern Sets (CoGAPS) (Fertig et al.,2010) is one such Bayesian decomposition NMF algorithm which computes non-orthogonal basis vectors of the latent space.

To compute the projection of target data on latent space calculated from source dataset projectR uses multiple linear regressions. This essentially calculates orthogonal projections of the data on the latent space. Given the target data matrix $T$ and amplitude matrix $A$, the projection is computed as

$$T_j \sim A\beta_j \tag{1.4}$$

As shown in equation 1.4, the $j^{th}$ column of the target data matrix is regressed on $A$ to generate the projected pattern weights for each target dataset sample. The final projected matrix is samples by patterns. Thus, the patterns are transferred to the target samples.

### 1.4.3 K-means clustering

K-means clustering assigns each data point to one of the k clusters. To prepare it for projection, the data needs to be transformed to generate a loadings matrix. The number of columns in the loadings is equal to k and can be considered as number of patterns similar to NMF. The pattern weights for each cluster are assigned by calculating the correlation between the values for each gene present in the cluster and average value for all samples in the cluster. This generates the loadings matrix. The projection of the target data matrix on loadings matrix is calculated exactly as NMF.

### 1.4.4 Hierarchical clustering

The approach to compute projection for hierarchical clustering is similar to k-means clustering. The first step is to generate k clusters using `cutree` function in R. After generating the clusters, method specified in section 1.4.3 is used.

## 1.5 P-value computations

### 1.5.1 Wald test

Since the projections are calculated from linear regressions, for each projected pattern weight $\hat{\beta}_{ij}$, we can test the null hypothesis $H_0$: $\hat{\beta}_{ij} = \hat{\beta}_0$ the p-value can be computed using wald-test. The wald statistic is calculated as follows:

$$w = \frac{\hat{\beta}_{ij} - \hat{\beta}_0}{se(\hat{\beta}_{ij})}$$

where $se(\hat{\beta}_{ij})$ is the standard error in the estimate. The statistics follow a student-t distribution with n – p degrees of freedom, where n is the number of samples in the target data and p is the number of patterns. A p-value can be calculated using the statistics.

### 1.5.2 Bootstrap test

Another method to calculate the p-values for the estimated projected pattern weights that is implemented in projectR generates sampling distribution of the weights using bootstrap. Based on percentile confidence interval approach (Efron, B. 1987), 100 (1 – α) percent confidence interval for the estimates, quantiles from α to 1 – α are used, where α is the significance level. To calculate the p-value for the null hypothesis $H_0$: $\hat{\beta}_{ij} = 0$ , the following method is used:

$$\text{p-value} \begin{cases} 2 \times \text{quantile}(0), & \text{if quantile}(0) \leq 0.5 \\ 2 \times (1 - \text{quantile}(0)), & \text{if quantile}(0) > 0.5 \end{cases}$$

# Chapter 2

# Implementation and application

This chapter deals with the implementation and an application case study of projectR. The implementation of transfer learning using projectR uses S4 object-oriented programming framework of R.

## 2.1 Functions

The main function of the package is `projectR` which is a S4 generic function. The implementation of the function is different based on the input. The important arguments to the function are `data, loadings, bootstrapPval,` and `bootIter` which are for target data matrix, latent space learned on source data, logical to execute p-values using bootstrap approach and number of bootstrap iterations respectively. The supported classes of latent space are `prcomp` (PCA), `hclust` (hierarchical clustering), `kmeans` (k-means clustering), `CogapsResults` (CoGAPS) and `matrix` (for all other methods).

A key function is `geneMatchR` which subsets target dataset and source data latent space to include only common genes. Another function `cluster2pattern` to covert `hclust` and `kmeans` to patterns.

An important aspect of transfer learning is identifying patterns that are associated with data annotations. After the projections are computed using projectR, `aucMat` can be used to compute AUC (area under the curve) values for prediction of annotations from the projected values using `performance` and `prediction` functions from ROCR.

Visualization of patterns associated with data labels can be done using `alluvialMat` function provided in the package. The association is decided based on corrected p-values of the projected pattern weights.

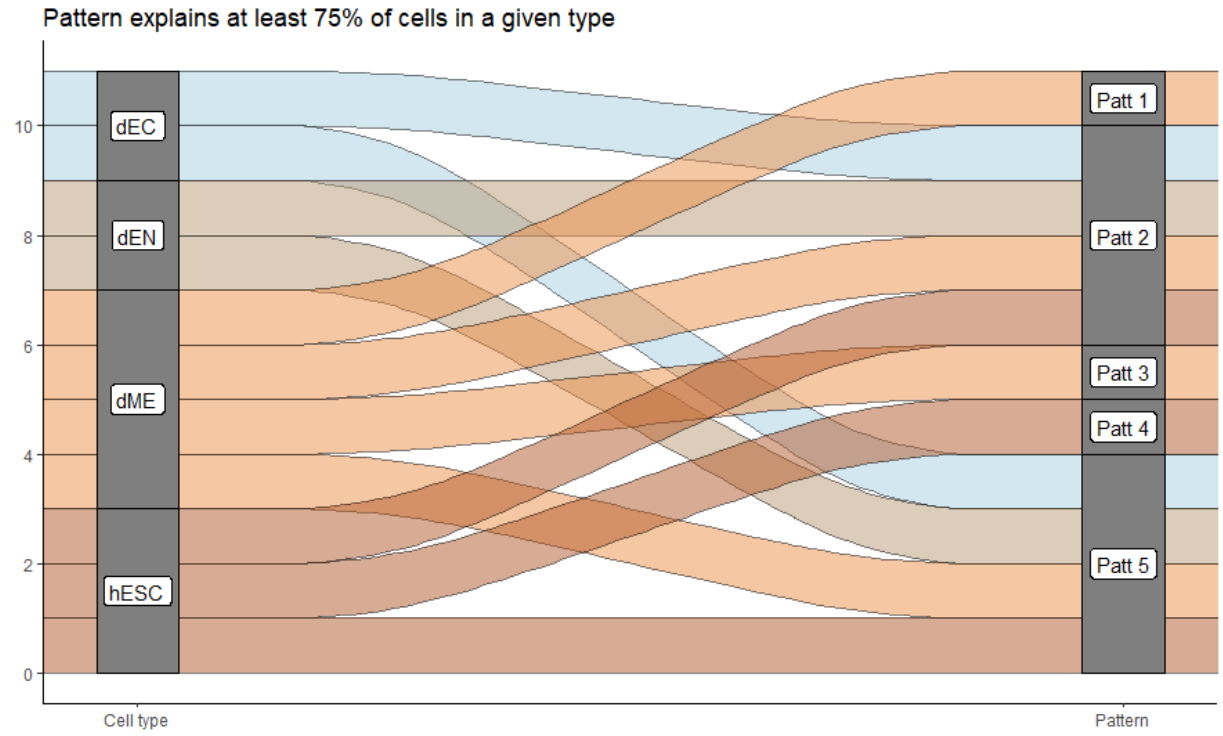


Figure 2.1: Alluvial plot generated using projectR

## 2.2 Workflow

Figure 2.1 shows the projectR workflow. The first step is computing the latent space(s). Projection of target data set is the next step. There can be several target datasets for one source dataset. Different datasets can be selected based on different aspects of shared biology.

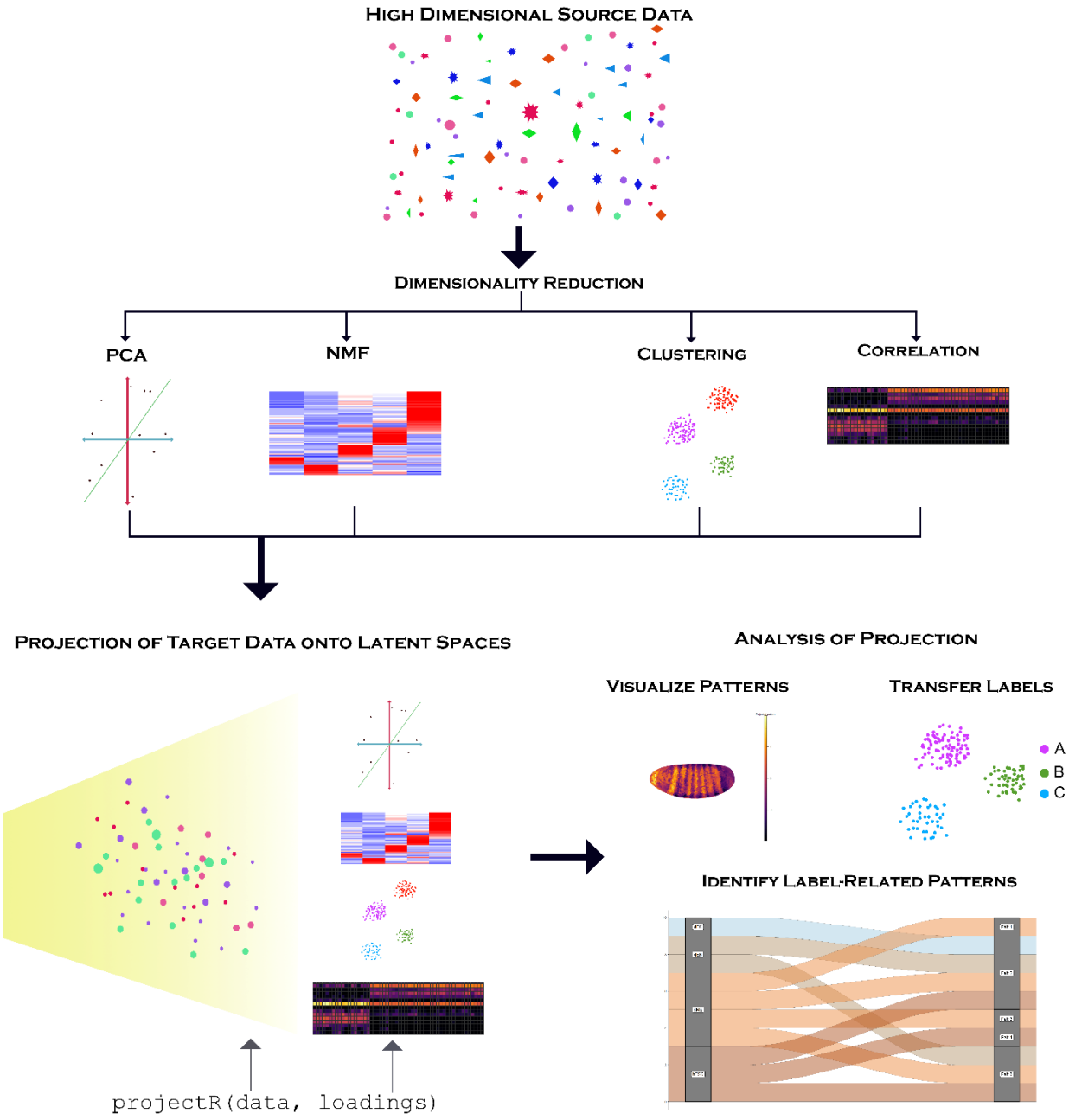


Figure 2.2: projectR workflow

For instance, if the source dataset is single-cell RNA-sequencing of different sections of mouse brain collected on different days of development, the target datasets could be same sections of mouse collected in an independent experiment, brain sections from human or other species, sections from other regions of the central nervous system such as retina (Stein-O'Brien et al.

2019), data from different technology such as ATAC-Seq, etc (Erbe et al. 2020). The analysis of projected pattern is the subsequent step.

## 2.3 Application

A single-cell RNA sequencing dataset of 1297 cells from stage 6 *Drosophila* embryo was used as the source dataset (Karaiskos et al. 2017). At this stage, the embryo has about 6000 cells. Drosophila development model is well studied and thus the results generated from using projectR in this context could be easily verified. During development multiple genes show spatially restrictive gene expression meaning these genes are expressed only in specific spatial locations on the embryo. 20 CoGAPS patterns were learned from the source dataset. For this stage of development, expression of 84 genes can be used to identify the spatial location of a cell. A binarized matrix of 84 genes and 3039 symmetrical positions on the embryo was generated from *in-situ* imaging data. This matrix was used as target data matrix. Projecting it on the latent space learned on the source dataset provided the spatial distribution of the learned patterns. The patterns were visualized in 3D. Figure 2.3 shows a 2D snapshot of the visualization.

Notably, the patterns are learned from the transcriptomics data and transferred to spatial coordinates. Such spatial visualization of transcriptomics patterns is not possible with either of the dataset alone. This demonstrates that projectR is useful in integrative analysis. However, this may not reveal all the spatial patterns in the *Drosophila* embryo and not all patterns may be spatially meaningful, i.e, they may not correspond to biologically relevant spatial distribution of gene expression.

To validate the spatial patterns, genes with high weights for a pattern were identified and the spatial pattern was compared with their virtual *in situ* hybridization (vISH) distribution computed using DistMap (https://github.com/rajewsky-lab/distmap) (Karaiskos et al. 2017). Pattern 13 was the most prominent spatial pattern as defined by visual gene expression and quantitative

validation. Quantitative validation was done by calculating the correlations between vISH gene expression and spatial pattern. Most of the genes were highly correlated, with correlation coefficient > 0.9. Table 2.1 shows the top genes with highest weights for pattern 13 and their correlations.

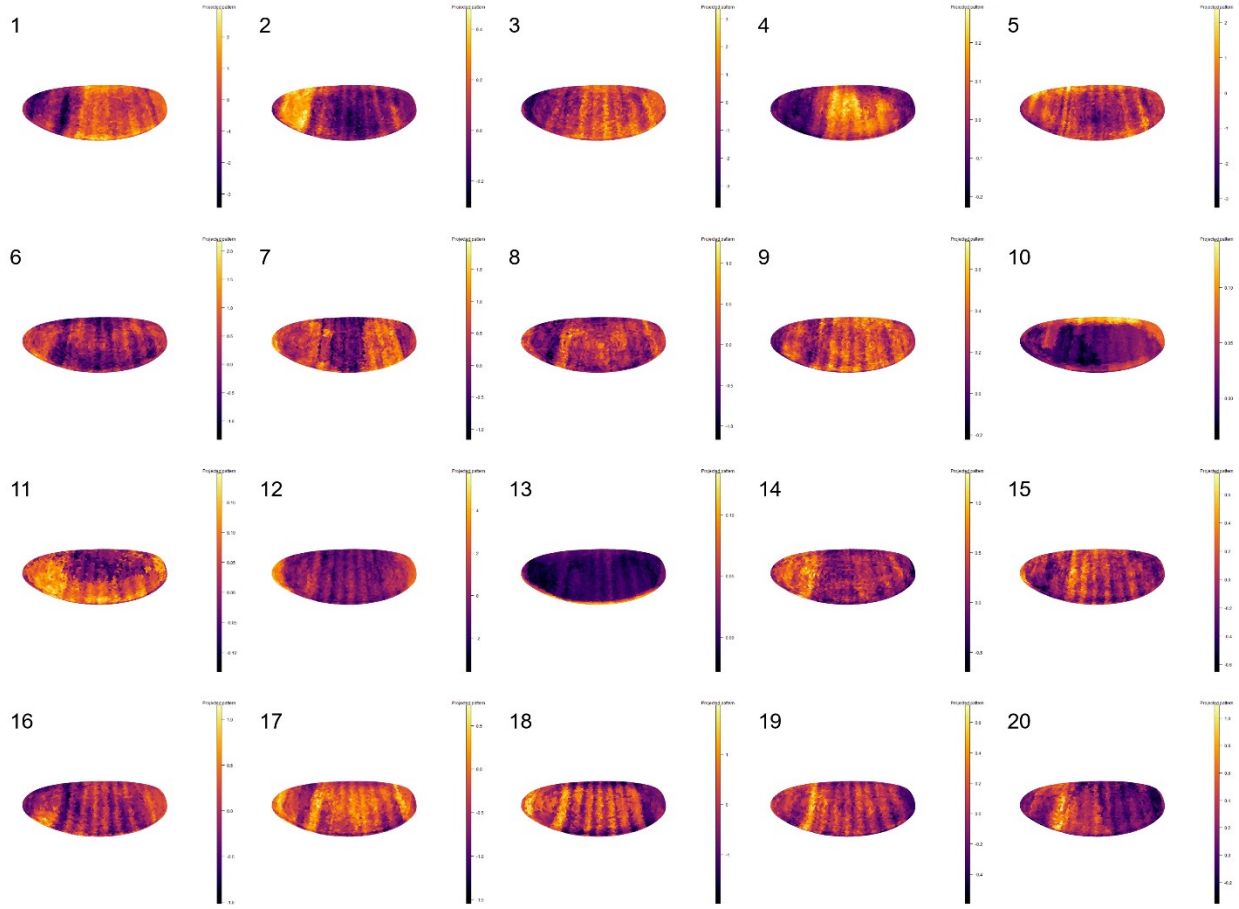


Figure 2.3: Spatial visualization of projected patterns. The on top left is the pattern number. This is a lateral view with anterior on left and posterior on right side. The scale on the right is the pattern weight.

| Gene | Correlation |
|---|---|
| **Ilp4** | 0.924 |
| **twi** | 0.912 |
| **Cyp310a1** | 0.92 |
| **ventrally-expressed-protein-D** | 0.902 |
| **CG4500** | 0.895 |
| **Ama** | 0.698 |
| **CG12177** | 0.915 |
| **sna** | 0.906 |
| **Mes2** | 0.915 |
| **CG3036** | 0.912 |

Table 2.1: Top genes associated with pattern 13 and correlations between their vISH values and pattern 13 weights.

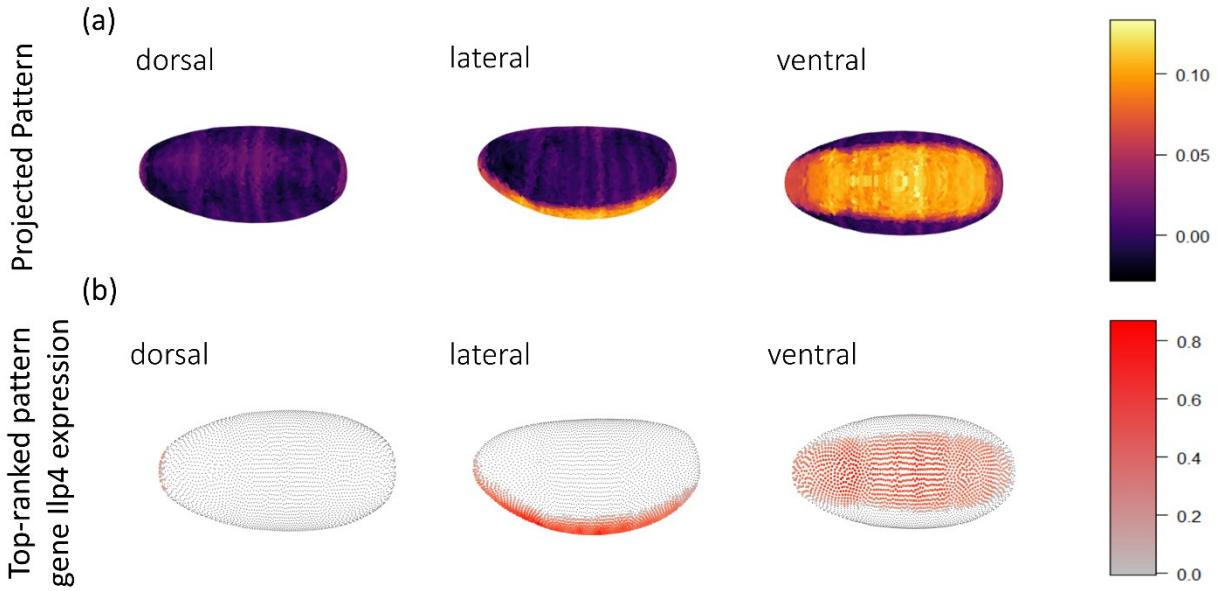


Figure 2.4: Visual comparison of spatial pattern 13 with vISH of gene Ilp4

The pattern shown in figure 2.4 is ventral and the spatial expression of the genes is also predominantly ventral. This demonstrates that not only CoGAPS was able to identify this spatial pattern, but also projectR enabled accurate visualization of the pattern.

## 2.3.1 Robustness of projection

The projection of the binarized *in-situ* matrix uses 84 genes for projection. To check if the projections are robust, a leave-one-out analysis was done. In this analysis, each gene was left out and projections were calculated. With each projection, a correlation between the left-out projection and projection with 84 genes was calculated. The analysis is visualized as a heatmap as shown in figure 2.5. The projections were found to be robust since the mean correlation value is 0.98 and more than 80% of correlation values are 0.99.

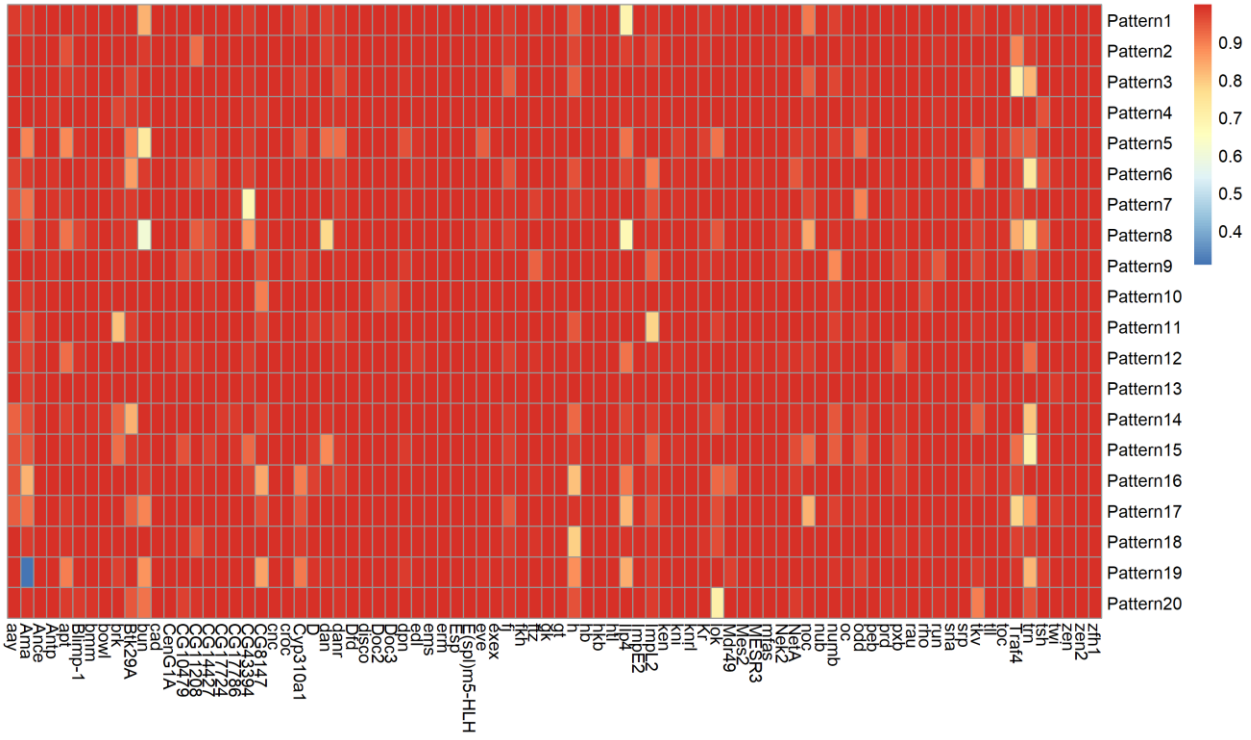


Figure 2.5: Leave-one-out analysis for the projections. Each column denotes the gene that was left-out in the projection. Each value is correlation between the projected pattern without a gene and with all genes.

## 2.4 Discussion

Transfer learning method projectR as a software package enables investigation of datasets using related dataset via dimension reduction. Several use cases of transfer learning have been mentioned earlier such as interpretation of the latent space dimensions, transferring annotations across datasets, validation of existing annotations, and gaining integrative understanding of the datasets. The application described in chapter 2 demonstrates the ability of transfer learning to integrate different data modalities, in this case — transcriptomics and spatial. This suggests that projectR can potentially be a useful tool in multi-omics analysis to integrate different data modalities. The current framework allows for transfer learning via a fixed number of dimension reduction methods, which can certainly be expanded to include a wider variety of methods used in different biological applications. Furthermore, since projectR uses different dimension reduction methods, the efficacy of transfer learning based on these methods need to be evaluated. Future work will involve assessment and improvement of projectR to enable better integrated multi-omics analysis which, for example, can include assignment of spatial coordinates to cells based on transcriptomics data.

# Bibliography


1. Efron, B. (1987). Better Bootstrap Confidence Intervals. Journal of the American Statistical Association. Journal of the American Statistical Association, 82 (397): pp. 171–185
2. Erbe, R., Kessler, M., Favorov, A., Easwaran, H., Gaykalova, D. and Fertig, E., 2020. Matrix factorization and transfer learning uncover regulatory biology across multiple single-cell ATAC-seq data sets. Biorxiv.
3. Fertig, E.J., Ding, J., Favorov, A.V., Parmigiani, G. and Ochs, M.F. 2010. CoGAPS: an R/C++ package to identify patterns and biological process activity in transcriptomic data. Bioinformatics 26(21), pp. 2792–2793.
4. Karaiskos, N., Wahle, P., Alles, J., et al. 2017. The Drosophila embryo at single-cell transcriptome resolution. Science 358(6360), pp. 194–199.
5. Meng, C., Zeleznik, O., Thallinger, G., Kuster, B., Gholami, A. and Culhane, A., 2016. Dimension reduction techniques for the integrative analysis of multi-omics data. Briefings in Bioinformatics, 17(4), pp.628-641.
6. Sharma, G., Colantuoni, C., Goff, L.A., Fertig, E.J. and Stein-O'Brien, G. 2020. projectR: An R/Bioconductor package for transfer learning via PCA, NMF, correlation, and clustering. Bioinformatics.

7. Stein-O’Brien, G.L., Clark, B.S., Sherman, T., et al. 2019. Decomposing Cell Identity for Transfer Learning across Cellular Measurements, Platforms, Tissues, and Species. Cell Systems 8(5), pp. 395-411.e8
8. Stein-O’Brien, G.L., Arora, R., Culhane, A.C., et al. 2018. Enter the Matrix: Factorization Uncovers Knowledge from Omics. Trends in Genetics 34(10), pp. 790–805.

# Biographical Statement

Gaurav Sharma graduated with a Bachelor's of Technology (B.Tech) in mechanical engineering and a minor in management from Indian Institute of Technology Gandhinagar in 2016. He has been awarded JN Tata Scholarship, Debesh-Kamal Scholarship, and Saryu Doshi Postgraduate Fellowship. He joined Master's of Science and Engineering at Johns Hopkins University in 2018. During his tenure at Johns Hopkins University, he published a Bioconductor package called projectR, and accompanying article in Bioinformatics. He has worked with single-cell analysis and co-authored another article published in the British Journal of Cancer.